typedef struct timeval bruh;

double timeval2ms(bruh b) {

    return b.tv\_usec / 1000.0 + b.tv\_sec \* 1000.0;

}

pthread\_t\* tid;

TID שומר על הזהויות של היוטים. ARGS ישמור על הארגומנט שישלח להם וTURN מסמן של מי נתור היה/יהיה. לשים לב כי האיתחול ב-1 ובנוסף לזה יש שני TIMEVALS.

int\* args;

int turn = -1;

bruh startTimer, endTimer;

void\* tf(void\* arg) {

בתחילת החוט יש שמירת זהות.

    int id = \*((int\*)arg);

    printf("ID: %d \n", id);

בלולאה אינסופית, יש בדיקה של שני מקרים: אם זה התור של אותו חוט או לא. במידה והתור הוא **לא** של אותו חוט ספציפי, הוא בודק אם הערך הוא לא מינוס. אם אכן לא מינוס, הוא שומר את הזמן הנוכחי והחוט שהיה תורו ומדפיס הפרש כך בחוט 2 יודפס זמן מעבד של חוט 1!!! חשוב לציין כי כמעט מייד לאחר דגימת זמן, התור מוחלף בהוא עצמו (ולכן בהכרח חייב לצאת מהלולאה).

    while(1) {

        while(turn != id) {

            if(turn == -1) {

                turn = id;

                break;

            }

            else{

                gettimeofday(&endTimer, NULL);

                int temp = turn;

                turn = id;

                double etime =

timeval2ms(endTimer) - timeval2ms(startTimer);

                printf("Elapsed Time for thread %d is %f\n", temp, etime);

כל עוד זה התור של אותו חוט, החוט פשוט עומד ומעדכן את זמן ההתחלה.

            }

        }

        while(turn == id) gettimeofday(&startTimer, NULL);

    }

}

Elapsed Time for thread 1 is 0.000000

Elapsed Time for thread 1 is 0.004883

Elapsed Time for thread 0 is 0.000000

Elapsed Time for thread 0 is 0.000000

Elapsed Time for thread 0 is 0.005859

Elapsed Time for thread 1 is 0.000000